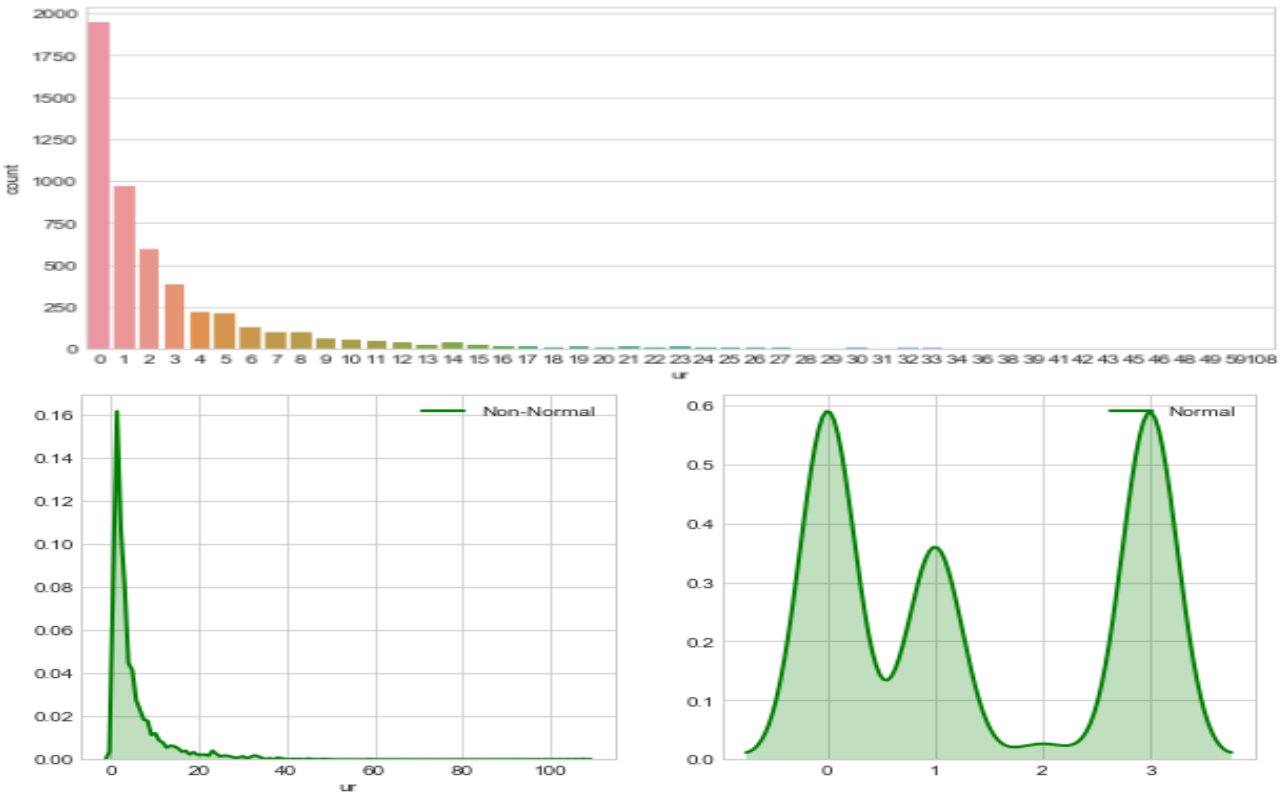
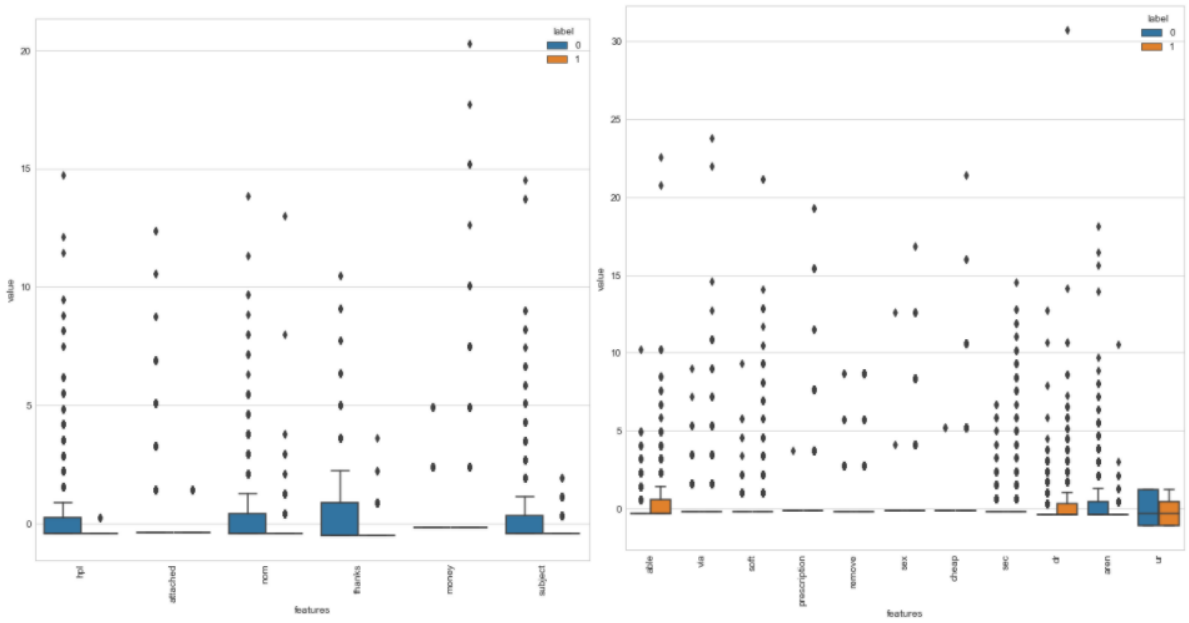
DATA MİNİNG PROJECT REPORT

Bu projede veri seti incelemesi, veri seti dengeli hale getirme, görselleştirmeler, outlier/anomaly tepitleri, feature normalization LDA, Cross Validation ve birkaç çeşit model kullanılmıştır.
Veri seti kaggle sitesinden alınmıştır,
Link: https://www.kaggle.com/balaka18/email-spam-classification-dataset-csv
Veriseti incelemesi ve veri setini tanımaya yönelik fonksiyonların kullanılması vardır. Bunlar .info(), .describe() .head() fonksiyonları ile yapılmıştır. Feature'lar kelimelerdir ve değerleri onların belirli bir email de kaç kez geçtiğini belirtiyor. 1 kategorical ve 3 bin numerik feature vardır. 1 tane de Prediciton adlı target feature'ı bulunmaktadır. Target feature 0 ise email spam değil, 1 ise spamdir. Yani Binary Classification için olan bir verisetidir. «Email No.» kategorical feature'ı unique ve Prediction için gereksiz bir özellik olduğu için verisetinden çıkarılmıştır.
Random forest algoritması kullanılarak 3 bin tane olan feature'ların skorları bulunmuştur bunların yaklaşık 1290 kadarı 0 skora sahiptir bu featurelar elenerek feature selection yapılmıştır. Geriye kalan featureların target variable ile aralarındaki korelasyon değerleri bulunmuştur. Maximum korelasyonlar 0.22 ve -0.27 olarak bulunmuştur. 0.15 ve -0.15 arasındaki korelasyon değerlerini iptal edilmek için yeterince az olduğuna karar verdim ve 1688 tane feature böylece iptal edilmiş oldu. Geriye 23 tane feature kaldı.
23 tane feature'ın heat map'ini çıkardım. Burada birbirleriyle yüksek korelasyonlu oaln featurelardan birini iptal ettim çünkü target variable üzerinde neredeyse aynı etkiye sahip fazladan feature'ların olması anlamsızdı ve ileride eğitilecek modellerde vakit kaybına ve overfitting'e sebep olabilirlerdi. 0.6 korelasyon değerini eşik değer oalrak (threshold) aldım, bu ve üzerindeki değerler iptal edilmek için yeterince yüksek olarak göz önünde bulunduruldu. Bu işlemden sonra 17 feature ve 1 target variable'ım kaldı.
Feature'ların dağılım şeklini görebilmek için count plot ile görselleştirildi. 'ur' feature'ı sola doğru skewed olduğu gözlemlendi ve normalizasyon uygulanarak dağılım görüntüsü değiştirildi.



Şekil 1 Skwewd Feature Normalization

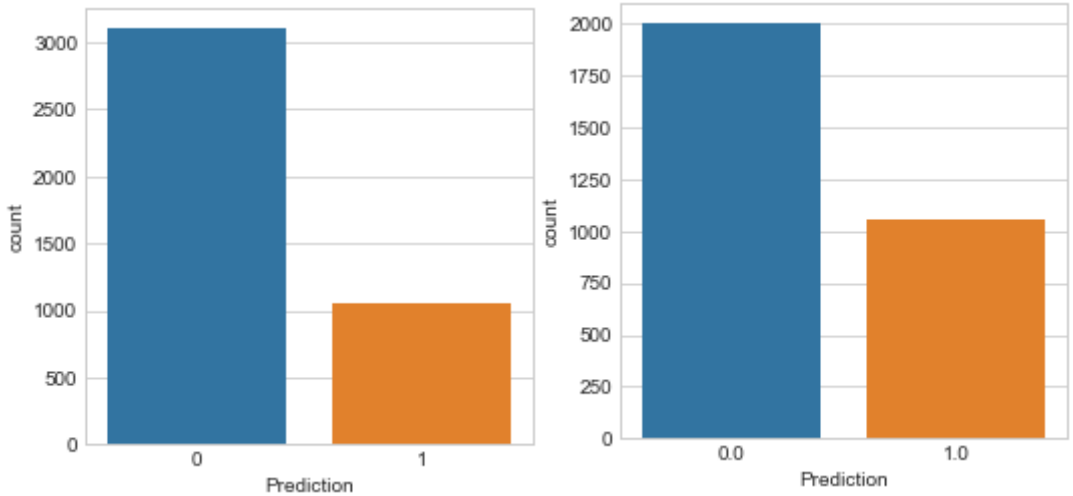
Veriseti Standard scaler ile ölçeklendirildi. Outlier'lar quartile yöntemi ile tespit edilip, verisetinden çıkarıldı. Outlierlar box plot ile görselleştirilip incelendi. Aynı zamanda 0 classına ait outlierlar 3 önemli feature'ın cluster yapılması ile ortaya çıkmış ve bunlar da veriseti dengeleştirilirken silinmiştir. Diğer count plotlar da ödevde bulunuyor raporu uzun yapmamak için hepsini koymadım.



Şekil 2 Box Plot Visualization

Veri seti dengeli mi ddeğil mi kontrolü için count plot ile sınıfların eleman sayılarını çizdirip grafik haline getirdim. Veri seti dengesiz (Unbalanced) olduğu ortaya çıktı.

Dengesiz Veriseti Dengelenen Veriseti



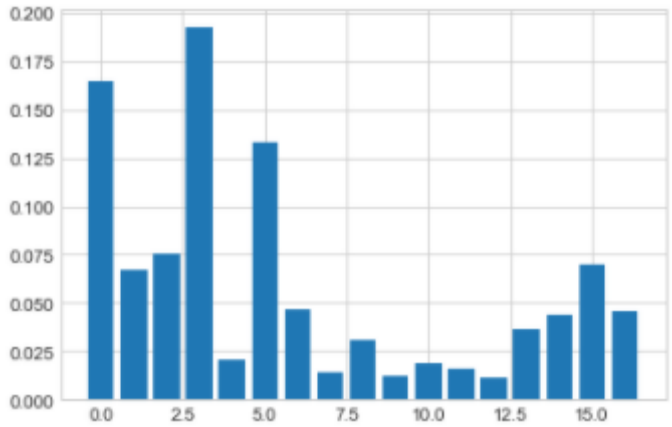
Şekil 3 Unbalanced Dataset Count Plot

,

Verisetini dengeli hale getirebilmek için https://ieeexplore.ieee.org/Xplore/home.jsp sitesinden birkaç makale okundu.
Bu makalelerin isimleri şunlardır:
->DBCS: Density based cluster sampling for solving imbalanced classification problem
->DBSM: The combination of DBSCAN and SMOTE for imbalanced data classification
->An under-sampling technique for imbalanced data classification based on DBSCAN algorithm
Undersampling için target variable ile arasında en büyük skor olan feature'ları alcağaım. Bunun için random forest gibi güçlü bir algoritmayı kullanarak backward elimination yaparak en fazla skorları olan 3 tane feature seçeceğim.

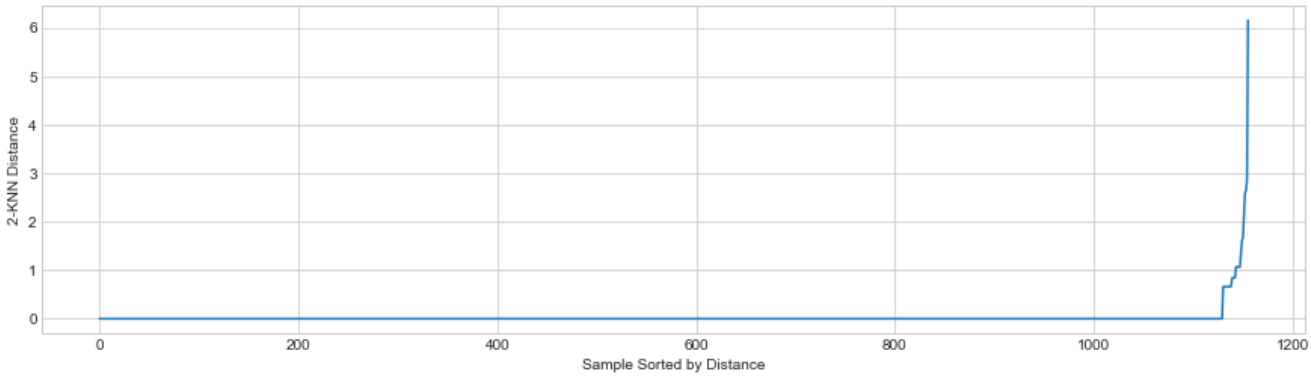
Best featurelar bunlar çıktı: `Out[156]: ['thanks', 'hpl', 'nom']`

Random Forest Algoritmasını Backward Elimination'da seçme sebebim de bu grafiktir:



Şekil 4 Features Score Graph

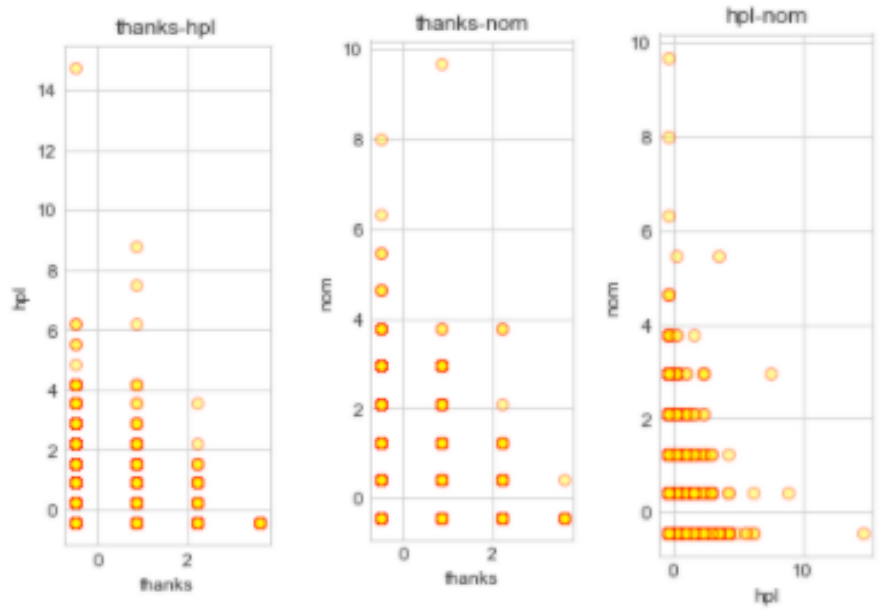
Sadece Prediciton değeri 0 olan objeleri alacağım, daha sonra bu 3 özellik ve target variable'ı alacağım yeni bir dataframe oluşturacağım. Bu 3 özelliği Unsupervised Clustering yaparak 3 farklı cluster oluşturdum. Bu clusterların büyüklüğü oranında içlerinden random elemanlar seçeceğim Toplamda 2 bin eleman elde etmek hedefim çünkü sınıfı 1 olan eleman sayısı bin tanedir ve dengesiz verisetinde grafikte de olduğu gibi sınıfı 0 oalnalrın 3 bin tane elemanı vardı, doğallığı bozmamak adına sayıyı bin değil 2 bin taneye düşüreceğim.
DBSCAN algoritması epsilon değeri için elbow metodu kullanıldı.
Son bin sayıya zoom yapıldı ki elbow ile bulunacak değer doğru şekilde görünebilsin diye:



Şekil 5 Elbow method 3 best feature dataframe for 0 prediction values

eps=1.2 min point=90

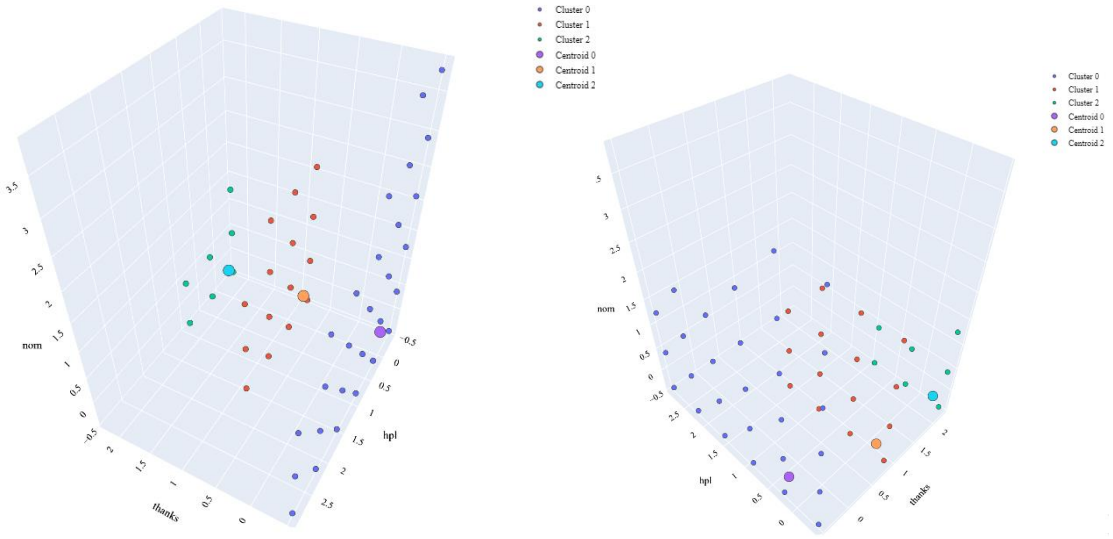
Veri seti dengeleştirilmesinde kullanılan 3 özellik ve grafikleştirilmesi.



Şekil 6 plot 3 combination of best features to observe datas behaviour

Cluster yapılan 3 feature'ın birbiryle olan kombinasyonlarının grafiği

Gördüğümüz koyu renkli noktalar birçok noktanın üst üste gelmesi ile oluşmuştur, normalde her nokta grafikteki sönük noktalar gibidir. Üst üste olma durumları görünmesi için alpha değeri kullanılmıştır.



Şekil 7 Cluster Result and Their Centroids

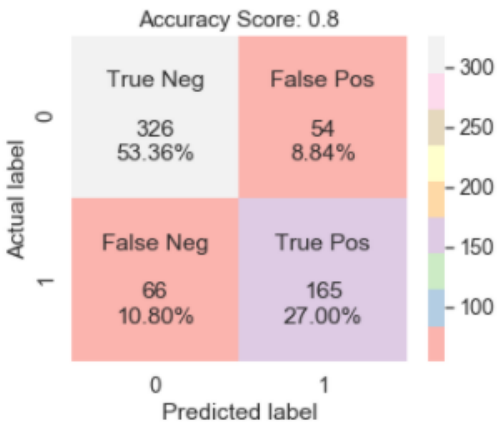
Her bir clusterdaki eleman sayısı tüm clusterlarınkine bölünerek oranları bulunmuştur ve şöyledir: [0.6807730756414528, 0.2785738087304232, 0.04065311562812396]

Bu oranlara göre 2000 tanenin %68 tane elemanı ilk clusterdan, %28 tanesi ikinci clusterdan ve kalanı da üçüncü clusterdan random olarak alınmıştır. Bu oranın sebebi çeşitliliğin ve doğal yapının bozulmaması için yapılmıştır.
LDA tekniği kullanımı : LDA feature reduction tekniği binary classification gerektiren veri setlerine çok uygun bir tekniktir. Bu teknik uygulanıp bir tek LDA feature'ı elde edilmiştir.
Bu feature ve Prediciton target değişkeni kullanılarak, Logistic regresyon uygulanmıştır. Aynı zamanda LDA uygulamadan direkt veri seti üzerinde Logistic regresyon uygulanarak sonuçlar karşılaştırılmıştır. LDA %2 daha az doğruluk oranı vermiştir ve bu veriseti için oldukça başarılı olduğu gözlemlenmiştir:

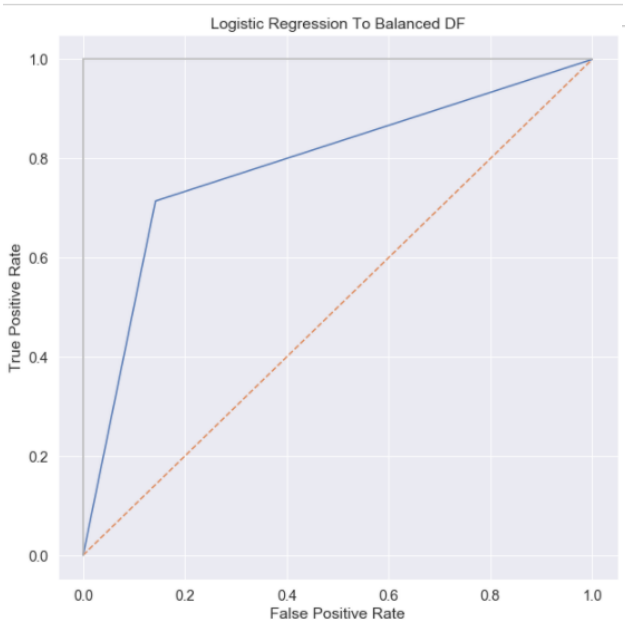
Solely Dataframe

Prediction Accuracy Without Cross Validation: 80.36%
Classification Report:

	precision	recall	f1-score	support
0.0	0.83	0.86	0.84	380
1.0	0.75	0.71	0.73	231
accuracy			0.80	611
macro avg	0.79	0.79	0.79	611
weighted avg	0.80	0.80	0.80	611



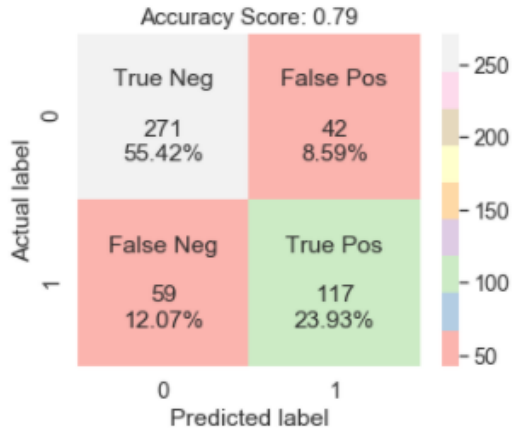
AUC: 0.7860902255639098



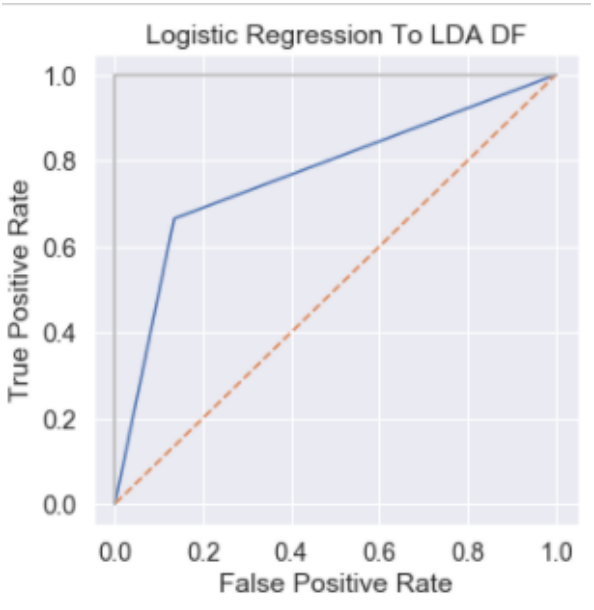
LDA Dataframe

Prediction Accuracy Without Cross Validation: 79.35%
Classification Report:

	precision	recall	f1-score	support
0.0	0.82	0.87	0.84	313
1.0	0.74	0.66	0.70	176
accuracy			0.79	489
macro avg	0.78	0.77	0.77	489
weighted avg	0.79	0.79	0.79	489



AUC: 0.7652937118791753

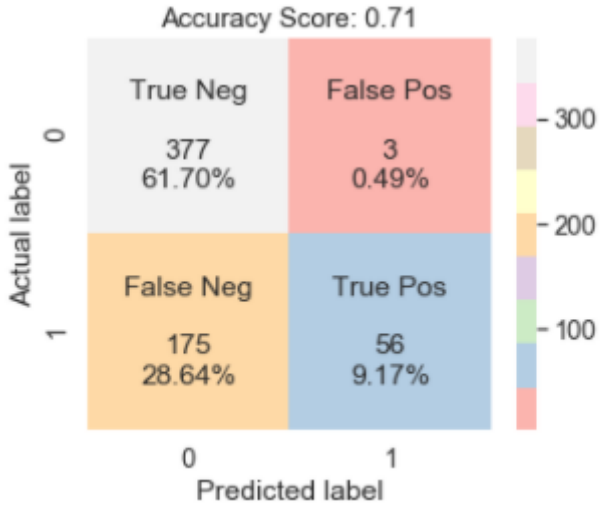


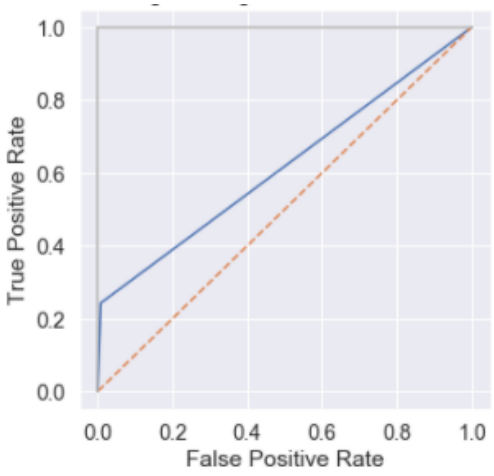
Naive Bayes Algoritması

```
Number of mislabeled points out of a total 832 points : 178
Prediction Accuracy Without Cross Validation: 70.87%
Classification Report:
              precision    recall  f1-score   support

         0.0       0.68      0.99      0.81       380
         1.0       0.95      0.24      0.39       231

    accuracy                           0.71       611
   macro avg       0.82      0.62      0.60       611
weighted avg       0.78      0.71      0.65       611
```






```
AUC: 0.6172647527910685
```

Apply 10 fold Cross Validation Result:

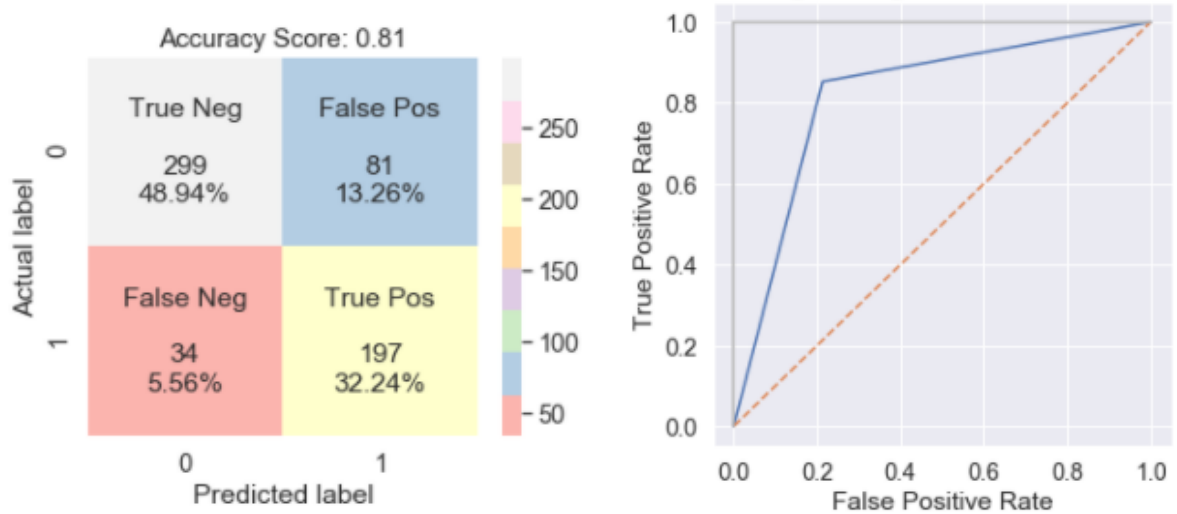
```
0.7484253723347262
```

Apply Decision Tree for Binary Classification

```
Prediction Accuracy Without Cross Validation: 81.18%
Classification Report:
              precision    recall  f1-score   support

         0.0       0.90      0.79      0.84       380
         1.0       0.71      0.85      0.77       231

    accuracy                           0.81       611
   macro avg       0.80      0.82      0.81       611
weighted avg       0.83      0.81      0.81       611
```




```
AUC: 0.8198279790385054
```

```
Scores: [0.89353774 0.89363208 0.83558962 0.89716981 0.85995283 0.92416667
 0.90195238 0.94452381 0.94788095 0.9132381 ]
Mean: 0.9011643980233602
Standard Deviation: 0.03296555404148409
```

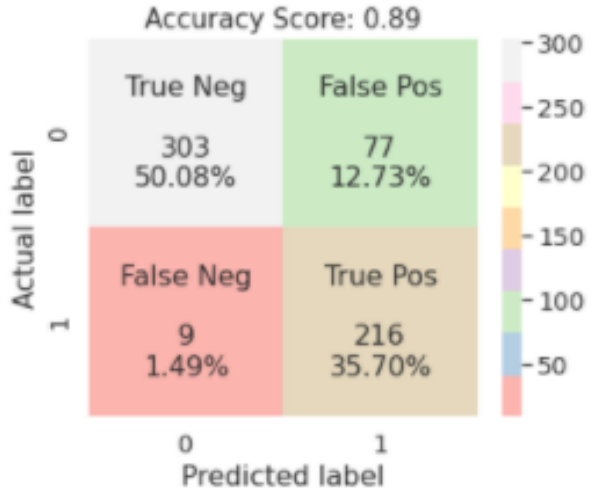
Apply Random Forest for Binary Classification

Applied 10 fold Cross Validation :

```
Prediction Accuracy Without Cross Validation: 89.49%
Classification Report:
              precision    recall  f1-score   support

         0.0       0.97      0.80      0.88       380
         1.0       0.74      0.96      0.83       225

    accuracy                           0.86       605
   macro avg       0.85      0.88      0.85       605
weighted avg       0.88      0.86      0.86       605
```




```
Scores: [0.8615534  0.8756068  0.86747573 0.86385922 0.83832524 0.86740196
 0.90877451 0.97879902 0.97127451 0.91583333]
Mean: 0.8948903721682848
Standard Deviation: 0.045452729491586186
```